METHOD AND MESSAGE THEREFOR OF MONITORING THE SPARE CAPACITY OF A DRA NETWORK

Cross Reference to Related Applications

5 ~~INSA~~ This application is a continuation-in-part of patent application serial number 09/038,531 filed on March 11, 1998, which claims priority under 35 U.S.C. § 119(e) to provisional application number 60/040536, filed March 12, 1997. The instant invention relates to

10 the following applications having serial No. 09/046,089 filed March 23, 1998, serial No. _____ (ALCA-1100-2) entitled "Restricted Reuse of Intact Portions of Failed Paths", serial No. _____ (ALCA-1100-7) entitled "Signal Conversion for Fault Isolation", and serial

15 No. _____ (ALCA-1100-6) entitled "Method and Message Therefor of Monitoring The Spare Capacity of a DRA Network". The respective disclosures of those applications are incorporated by reference to the disclosure of the instant application.

Field of the Invention

This invention relates to a distributed restoration algorithm (DRA) network, and more particularly to a method of monitoring the topology of the spare links in the network for rerouting traffic in the event that the traffic is disrupted due to a failure in one of the working links of the network.

Background of the Invention

In a telecommunications network provisioned with a distributed restoration algorithm (DRA), the network is capable of restoring traffic that has been disrupted due to a fault or malfunction at a given location thereof. In such DRA provisioned network, or portions thereof which are known as domains, the nodes, or digital cross-connect switches, of the network are each equipped with the DRA algorithm and the associated hardware that allow each node to seek out an alternate route to reroute traffic that has been disrupted due to a malfunction or failure at one of the links or nodes of the network. Each of the nodes is interconnected, by means of spans that include working and spare links, to at least one other node. Thus, ordinarily each node is connected to an adjacent node by at least one working link and one spare link. It is by means of these links that messages, in addition to traffic signals, are transmitted to and received by the nodes.

In a DRA network, when a failure occurs at one of the working links, the traffic is rerouted by means of the spare links. Thus, to operate effectively, it is required that the spare links of the DRA network be functional at all times, or at the very least, the network has a preconceived notion of which spare links are functional and which are not.

There is therefore a need for the instant invention DRA network to always have an up-to-date map of the functional spare links, i.e. the spare capacity,

of the network, so that traffic that is disrupted due
to a failure can be readily restored.

Summary of the Present Invention

To provide an up-to-date map of the functional spare links of the network, a topology of the network connected by the functional spare links is made available to the custodial nodes that bracket a malfunctioned link as soon as the failure is detected. The custodial node that is designated as the sender or origin node then uses the topology of the spare links to quickly reroute the traffic through the functional spare links.

To ensure that the spare links are functional, prior to the DRA process, special messages, referred to in this invention as keep alive messages, are continuously exchanged on the spare links between adjacent nodes. Each of these keep alive messages has a number of fields which allow it to identify the port of the node from which it is transmitted, the identification of the node, the incoming IP address and the outgoing IP address of the node, as well as a special field that identifies the keep alive message as coming from a custodial node when there is a detected failure. These keep alive message may be transmitted over the C-bit channels as idle signals.

So long as a spare link is operating properly, the keep alive messages that traverse therethrough will contain data that informs the network, possibly by way of the operation support system, of the various pairs of spare ports to which a spare link connects a pair of adjacent nodes. This information is collected by the

6

network and constantly updated so that at any moment,
the network has a view of the entire topology of the
network as to what spare links are available. This
data can be stored in a database at the operation

5   support system of the network, so that it may be
provided to the origin node as soon as a failure is
detected.

It is therefore an objective of the present
invention to provide a method of mapping a topology of

10   the spare capacity of a DRA network so that traffic may
be routed through the functional spare links when a
failure occurs at the network.

It is another objective of the present invention
to provide a special message that is exchanged

15   continuously between adjacent nodes before the
occurrence of the failure in order to continually
collect data relating to the available spare links of
the network.

Brief Description of the Figures

The above mentioned objectives and advantages of the present invention will become more apparent and the invention itself will be best understood by reference to the following description of an embodiment of the invention taken in conjunction with the accompanying drawings, wherein:

FIGURE 1 conceptually illustrates a simplified telecommunications restoration network to provide certain definitions applicable to the present invention;

FIGURE 2 illustrates a restoration subnetwork for illustrating concepts applicable to the present invention;

FIGURE 3 conceptually shows a failure within a restoration subnetwork;

FIGURE 4 illustrates two origins/destination nodes pairs for demonstrating the applicable scope of the present invention;

FIGUREs 5A and 5B illustrate the loose synchronization features of the present invention;

FIGURE 6 shows the failure notification message flow applicable to the present invention;

FIGURE 7 illustrates the flow of keep-alive messages according to the present invention;

FIGURE 8 illustrates the flow of path verification messages according to the teachings of the present invention;

8

FIGURE 9 shows a time diagram applicable to the failure notification and fault isolation process of the present invention;

FIGUREs 10 and 11 illustrate the AIS signal

5   flow within the restoration subnetwork of the present invention;

FIGURE 12 describes more completely the failure notification message flow within the restoration subnetwork according to the present invention;

10   FIGURE 13 illustrates the beginning of an iteration of the restoration process of the present invention;

FIGURE 14 provides a timed diagram applicable to the explore, return, max flow and connect phases of the

15   first iteration of the restoration process of the present invention;

FIGURE 15 provides a timed diagram associated with the explore phase of the process of the present invention;

20   FIGURE 16 illustrates the possible configuration of multiple origins/destination node pairs from a given origin node;

FIGURE 17 depicts two steps of the explore phase of the first iteration of the restoration process;

25   FIGURE 18 provides a timed diagram applicable to the return phase of the restoration process of the present invention;

FIGURE 19 shows steps associated with the return phase of the present process;

FIGUREs 20, 21 and 22 illustrates the link allocation according to the return phase of the present invention;

FIGURE 23 illustrates a typical return message for receipt by the origin node of a restoration subnetwork;

FIGURE 24 provides a timed diagram for depicting the modified map derived from the return messages received at the origin node;

FIGURE 26 shows the max flow output for the max flow phase of the present process;

FIGURE 27 illustrates an optimal routing applicable to the max flow output of the present invention;

FIGURE 28 provides a timed diagram for showing the sequence of the connect phase for the first iteration of the process of the present invention;

FIGURE 29 illustrates the connect messages for providing the alternate path routes between an origin node and destination node of a restoration subnetwork;

FIGUREs 30 and 31 show how the present invention deals with hybrid restoration subnetworks;

FIGUREs 32 and 33 illustrate the explore phase and return phase, respectively, applicable to hybrid networks;

FIGURE 34 shows the time diagram including an extra iteration for processing hybrid networks according to the teachings of the present invention;

FIGUREs 35 and 36 illustrate a lower quality spare according to the teachings of the present invention;

FIGURE 37 illustrate the use of a "I am custodial node" flag of the present invention;

FIGUREs 38 through 42 describe the restricted re-use features of the present invention;

FIGURE 43 describes the path inhibit feature of the present invention;

FIGURE 44 further describes the path inhibit feature of the present invention;

FIGURE 45 is an illustration of a telecommunications network of the instant invention;

FIGURE 46 is a block diagram illustrating two adjacent cross-connect switches and the physical interconnection therebetween; and

FIGURE 47 is an illustration of the structure of an exemplar keep alive message of the present invention.

Detailed Description of the Present Invention

FIGURE 1 shows telecommunications network portion 10, that includes node 12 that may communicate with node 14 and node 16, for example. Connecting between
5   node 12 and 14 may be a set of links such as links 18 through 26, as well as for example, links 28 through 30 between node 12 and node 16. Node 14 and node 16 may also communicate between one another through links 32 through 36, for example, which collectively may be
10   thought of as a span 38.

The following description uses certain terms to describe the concepts of the present invention. The term 1633SX is a cross-connect switch and is here called a "node." Between nodes are links, which may be
15   a DS-3, and STS-1, which is essentially the same thing as a DS-3, but which conforms to a different standard. A link could be an STS-3, which is three STS-1s multiplexed together to form a single signal. A link may also be a STS-12, which is twelve STS-1s
20   multiplexed together, or a link could be an STS-12C, which is twelve STS-12s, which are actually locked together to form one large channel. A link, however, actually is one unit of capacity for the purposes of the present invention. Thus, for purposes of the
25   following description, a link is a unit of capacity connecting between one node and another. A span is to be understood as all of the links between two adjacent nodes. Adjacent nodes or neighbor nodes are connected by a bundle, which itself is made up of links.

For purposes of the present description, links may
be classified as working, spare, fail, or recovered. A
working link is a link that currently carries traffic.
Spare links are operable links that are not currently
being used. A spare link may be used whenever the
network desires to use the link. A failed link is a
link that was working, but has failed. A recovered
link is a link that, as will be described more
completely below, has been recovered.

FIGURE 2 illustrates the conceptual example of
restoration subnetwork 40 that may include origin node
42 that through tandem nodes 44 and 46 connects to
destination node 48. In restoration subnetwork 40, a
path such as paths 50, 52, 54, and 56 includes
connections to nodes 42 through 48, for example, as
well as links between these nodes. As restoration
subnetwork 40 depicts, each of the paths enters
restoration subnetwork 40 from outside restoration
subnetwork 40 at origin node 42.

With the present embodiment, each of nodes 42
through 48 includes an associated node identifier.
Origin node 42 possesses a lower node identifier value,
while destination node 48 possesses a higher node
identifier value. In the restoration process of the
present invention, the nodes compare node
identification numbers.

The present invention establishes restoration
subnetwork 40 that may be part of an entire
telecommunications network 10. Within restoration

subnetwork 40, there may be numerous paths 50. A path
50 includes a number of links 18 strung together and
crossconnected through the nodes 44. The path 50 does
not start within restoration subnetwork 40, but may

5   start at a customer premise or someplace else. In
fact, a path 50 may originate outside a given
telecommunications network 10. The point at which the
path 50 enters the restoration subnetwork 40, however,
is origin node 42. The point on origin node 42 at

10   which path 50 comes into restoration subnetwork 40 is
access/egress port 58.

In a restoration subnetwork, the failure may occur
between two tandem nodes. The two tandem nodes on each
side of the failure are designated as "custodial"

15   nodes. If a single failure occurs in the network,
there can be two custodial nodes. In the network,
therefore, there can be many origin/destination nodes.
There will be two origin nodes and two destination
nodes. An origin node together with an associated

20   destination node may be deemed an origin/destination
pair. One failure may cause many origin/destination
pairs.

FIGURE 3 illustrates the concept of custodial
nodes applicable to the present invention. Referring

25   again to restoration subnetwork 40, custodial nodes 62
and 64 are the tandem nodes positioned on each side of
failed span 66.   Custodial nodes 62 and 64 have
bound the failed link

and communicate this failure, as will be described
below.   FIGURE 4 illustrates the aspect of the
present
invention for handling more than one origin-destination

5   node pair in the event of a span failure. Referring to
FIGURE 4, restoration subnetwork 40 may include, for
example, origin node 42 that connects through custodial
nodes 62 and 64 to destination node 48. Within the
same restoration subnetwork, there may be more than one

10   origin node, such as origin node 72. In fact, origin
node 72 may connect through custodial node 62 and
custodial node 64 to destination node 74. As in FIGURE
3, FIGURE 4 shows failure 66 that establishes custodial
nodes 62 and 64.

15   The present invention has application for each
origin/destination pair in a given restoration
subnetwork. The following discussion, however,
describes the operation of the present invention for
one origin/destination pair. obtaining an understanding

20   of how the present invention handles a single
origin/destination pair makes clear how the algorithm
may be extended in the event of several
origin/destination pairs occurring at the same time.
An important consideration for the present invention,

25   however, is that a single cut may produce numerous
origin/destination pairs.

FIGUREs 5A and 5B illustrate the concept of loose
synchronization according to the present invention.
"Loose synchronization" allows operation of the present

method and system as though all steps were synchronized according to a centralized clock. Known restoration algorithms suffer from race conditions during restoration that make operation of the restoration

5   process unpredictable. The restoration configuration that results in a given network, because of race conditions, depends on which messages arrive first. The present invention eliminates race conditions and provides a reliable result for each given failure.

10   This provides the ability to predict how the restored network will be configured, resulting in a much simpler restoration process.

Referring to FIGURE 5A, restoration subnetwork 40 includes origin node 42, that connects to tandem nodes

15   44 and 46. Data may flow from origin node 42 to tandem node 46, along data path 76, for example. Origin node 42 may connect to tandem node 44 via path 78. However, path 80 may directly connect origin node 42 with destination node 48. Path 82 connects between tandem

20   node 44 and tandem node 46. Moreover, path 84 connects between tandem node 46 and destination node 48. As FIGURE 5A depicts, data may flow along path 76 from origin node 42 to tandem node 46, and from destination node 48 to origin node 42. Moreover, data may be

25   communicated between tandem node 44 and tandem node 46. Destination node 48 may direct data to origin node 42 along data path 80, as well as to tandem node 46 using path 84.

These data flows will all take place in a single
step. At the end of a step, each of the nodes in
restoration subnetwork 40 sends a "step complete"
message to its neighboring node. Continuing with the

5   example of FIGURE 5A, in FIGURE 5B there are numerous
step complete messages that occur within restoration
subnetwork 40. In particular, step complete message
exchanges occur between origin node 42 and tandem node
44 on data path 78, between origin node 42 and tandem

10   node 46 on data path 76, and between origin node 42 and
destination node 48 on data path 80. Moreover, tandem
node 46 exchanges "step complete" messages with tandem
node 44 on data path 82, and between tandem node 46 and
destination node 48 on data path 84.

15   In the following discussion, the term "hop count"
is part of the message that travels from one node to
its neighbor. Each time a message flows from one node
to its neighbor, a "hop" occurs. Therefore, the hop
count determines how many hops the message has taken

20   within the restoration subnetwork.
The restoration algorithm of the present invention may
be partitioned into steps. Loose synchronization
assures that in each step a node processes the message
it receives from its neighbors in that step. Loose

25   synchronization also makes the node send a step
complete message to every neighbor. If a node has
nothing to do in a given step, all it does is send a
step complete message. When a node receives a step
complete message from all of its neighbors, it

increments a step counter associated with the node and
goes to the next step.

Once a node receives step complete messages from
every neighbor, it goes to the next step in the
restoration process. In looking at the messages that
may go over a link, it is possible to see a number of
messages going over the link. The last message,
however, will be a step complete message. Thus, during
the step, numerous data messages are exchanged between
nodes. At the end of the step, all the nodes send step
complete messages to their neighbors to indicate that
all of the appropriate data messages have been sent and
it is appropriate to go to the next step. As a result
of the continual data, step complete, data, step
complete, message traffic, a basic synchronization
occurs.

In practice, although the operation is not as
synchronized as it may appear in the associated
FIGUREs, synchronization occurs. During the operation
of the present invention, messages travel through the
restoration subnetwork at different times. However,
loose synchronization prevents data messages from
flowing through the restoration subnetwork until all
step complete messages have been received at the nodes.
It is possible for one node to be at step 3, while
another node is at step 4. In fact, at some 'places
within the restoration subnetwork, there may be even
further step differences between nodes. This helps

minimize the effects of slower nodes on the steps

occurring within the restoration subnetwork.

The steps in the process of the present invention

may be thought of most easily by considering them to be

5   numbered. The process, therefore, starts at step 1 and

proceeds to step 2. There are predetermined activities

that occur at each step and each node possesses its own

step counter. However, there is no master clock that

controls the entire restoration subnetwork. In other

10   words, the network restoration process of the present

invention may be considered as a distributive

restoration process. With this configuration, no node

is any different from any other node. They all perform

the same process independently, but in loose

15   synchronization.

FIGURE 6 shows the typical form of a failure

notification message through restoration subnetwork 40.

If, for example, origin node 42 desires to start a

restoration event, it first sends failure notification

20   messages to tandem node 44 via data path 78, to tandem

node 46 via data path 76, and destination node 48 via

data path 80. As FIGURE 6 further shows, tandem node

44 sends failure notification message to tandem node 46

on path 82, as does destination node 48 to tandem node

25   46 on path, 84.

The process of the present invention, therefore,

begins with a failure notification message. The

failure notification message is broadcast throughout

the restoration subnetwork to begin the restoration

process from one node to all other nodes. once a node receives a failure message, it sends the failure notification message to its neighboring node, which further sends the message to its neighboring nodes. Eventually the failure notification message reaches every node in the restoration subnetwork. Note that if there are multiple failures in a network, it is possible to have multiple failure notification messages flooding throughout the restoration subnetwork simultaneously.

The first failure notification message initiates the restoration algorithm of the present invention. Moreover, broadcasting the failure notification message is asynchronous in the sense that as soon as the node receives the failure notification message, it broadcasts the message to its neighbors without regard to any timing signals. It is the failure notification message that begins the loose synchronization process to begin the restoration process of the present invention at each node within the restoration subnetwork. Once a node begins the restoration process, a series of events occurs.

Note, however, that before the restoration process of the present invention occurs, numerous events are already occurring in the restoration subnetwork. One such event is the transmission and receipt of keep alive messages that neighboring nodes exchange between themselves.

FIGURE 7 illustrates the communication of keep-alive messages that the restoration process of the present invention communicates on spare links, for the purpose of identifying neighboring nodes. Referring to

5   FIGURE 7, configuration 90 shows the connection via spare link 92 between node 94 and node 96. Suppose, for example, that node 94 has the numerical designation I'll", and port designation 11103". Suppose further that node 96 has the numerical designation 3 and the

10 port designation 5. On spare link 92, node 94 sends keep-alive message 98 to node 96, identifying its node number "11" and port number "103". Also, from node 96, keep-alive message 100 flows to node 94, identifying the keep-alive message as coming from the node having

15 the numerical value "3", and its port having the numerical value "5".

The present invention employs keep-alive signaling using C-Bit of the DS-3 formatted messages in restoration subnetwork 40, the available spare links

20 carry DS-3 signals, wherein the C-bits convey special keep-alive messages. In particular, each keep-alive message contains the node identifier and port number that is sending the message, the WAN address of the node, and an "I am custodial node" indicator to be used

25 for assessing spare quality.

An important aspect of the present invention relates to signaling channels which occurs when cross-connect nodes communicate with one another. There are two kinds of communications the cross-connects can

perform. One is called in-band, another is out-of-
band. With in-band communication, a signal travels
over the same physical piece of media as the working
traffic. The communication travels over the same
physical media as the path or the same physical media
as the link. With out-of-band signals, there is
freedom to deliver the signals between cross-connects
in any way possible. Out-of-band signals generally
require a much higher data rate.

In FIGURE 7, for example, in-band messages are
piggybacked on links. out-of-band message traffic may
flow along any other possible path between two nodes.
With the present invention, certain messages must flow
in-band. These include the keep-alive message, the
path verification message, and the signal fail message.
There are some signaling channels available to the
restoration process of the present invention, depending
on the type of link involved. This includes SONET
links and asynchronous links, such as DS-3 links.

A distinguishing feature between SONET links and
DS-3 links is that each employs a different framing
standard for which unique and applicable equipment must
conform. It is not physically possible to have the
same port serve as a SONET port and as a DS-3 port at
the same time. In SONET signal channeling, there is a
feature called tandem path overhead, which is a
signaling channel that is part of the signal that is
multiplexed together. It is possible to separate this
signal portion from the SONET signaling channel.

Because of the tandem path overhead, sometimes called
the Z5 byte, there is the ability within the SONET
channel to send messages.

On DS-3 links, there are two possible signaling

5   channels. There is the C-bit and the X-bit. The C-bit
channel cannot be used on working paths, but can only
be used on spare or recovered links. This is because
the DS-3 standard provides the option using the C-bit
or not using the C-bit. If the C-bit format signal is

10   used, then it is possible to use the C-bit for
signaling. However, in this instance, working traffic
does not use that format. Accordingly, the C-bit is
not available for signaling on the working channels.
It can be used only on spare links and on recovered

15   links.

FIGURE 8 illustrates in restoration subnetwork 40
the flow of path verification messages from origin node
42 through tandem nodes 44 and 46 to destination node
48. Path verification message 102 flows from origin

20   node 42 through tandem nodes 44 and 46 to destination
node 48. In particular, suppose origin node 42 has the
label 18, and that working path 52 enters port 58.
Path verification message 102, therefore, contains the
labels 18 and 53, and carries this information through

25   tandem nodes 44 and 46 to destination node 48.
Destination node 48 includes the label 15 and egress
port 106 having the label 29. Path verification
message 104 flows through tandem node 46 and 44 to
origin node 42 for the purpose of identifying

destination node 48 as the destination node for working
path 52.

A path verification message is embedded in a DS-3
signal using the X-bits which are normally used for
5   very low speed single-bit alarm signaling. In the
present invention, the X-bit state is overridden with
short bursts of data to communicate signal identity to
receptive equipment downstream. The bursts are of such
short duration that other equipment relying upon
10   traditional use of the X-bit for alarm signaling will
not be disturbed.

The present invention also provides for confining
path verification signals within a network. In a DRA-
controlled network, path verification messages are
15   imbedded in traffic-bearing signals entering the
network and removed from signals leaving the network.
Inside of the network, propagation of such signals is
bounded based upon the DRA-enablement status of each
port. The path verification messages identify the
20   originating node and the destination node. The path
verification messages occur on working links that are
actually carrying traffic. The path verification
message originates at origin node 42 and the
restoration subnetwork and passes through tandem nodes
25   until the traffic reaches destination node 48. Tandem
nodes 44 and 46 between the origin node 42 and
destination node 48, for example, can read the path
verification message but they cannot modify it. At
destination node 48, the path verification message is

stripped from the working traffic to prevent its being transmitted from the restoration subnetwork.

The present invention uses the X-bit to carry path verification message 104. one signal format that the present invention may use is the DS-3 signal format. While it is possible to easily provide a path verification message on SONET traffic, the DS-3 traffic standard does not readily permit using path verification message 104. The present invention overcomes this limitation by adding to the DS-3 signal, without interrupting the traffic on this signal and without causing alarms throughout the network, path verification message 104 on the DS-3 frame X-bit.

The DS-3 standard specifies that the signal is provided in frames. Each frame has a special bit in it called the X-bit. In fact, there are two X-bits, X-1 and X-2. The original purpose of the X-bit, however, was not to carry path verification message 104. The present invention provides in the X-bit the path verification message. This avoids alarms and equipment problems that would occur if path verification message 104 were placed elsewhere. An important aspect of using the X-bit for path-verification message 104 with the present embodiment relates to the format of the signal. The present embodiment sends path verification message 104 at a very low data rate, for example, on the order of five bits per second. By sending path verification message 104 on the X-bit very slowly, the possibility of causing an alarm in the network is

significantly reduced. Path verification message 104 is sent at a short burst, followed by a long waiting period, followed by a short burst, followed by a long waiting period, etc. This method of "sneaking" path

5   verification message 104 past the alarms permits using path verification message 104 in the DS-3 architecture systems.

FIGURE 9 shows conceptually a timeline for the restoration process that the present invention

10   performs. With time moving downward, time region 108 depicts the network status prior to a failure happening at point 110. At the point that a failure happens, the failure notification and fault isolation events occur in time span 112. Upon completion of this step, the

15   first generation of the present process occurs, as indicated by space 114. This includes explore phase 116 having, for example three steps 118, 120 and 122. Return phase 124 occurs next and may include at least two steps 126 and 128. These steps are discussed more

20   completely below.

Once a failure occurs, the process of the present invention includes failure notification and fault isolation phase 112. Failure notification starts the process by sending failure notification messages

25   throughout the restoration subnetwork. Fault isolation entails determining which nodes are the custodial nodes. One reason that it is important to know the custodial nodes is that there are spares on the same span as the failed span. The present invention avoids

using those spares, because they are also highly likely

to fail. Fault isolation, therefore, provides a way to

identify which nodes are the custodial nodes and

identifies the location of the fault along the path.

5   FIGURE 10 illustrates the flow of AIS signals 130

through restoration subnetwork 40. In the event of

failure 66 between custodial nodes 62 and 64, the AIS

message 130 travels through custodial node 62 to origin

node 42 and out restoration subnetwork 40. Also, AIS

10   message 130 travels through custodial node 64 and

tandem node 46, to destination node 48 before leaving

restoration subnetwork 40. This is the normal way of

communicating AIS messages 130. Thus, normally every

link on a failed path sees the same AIS signal.

15   FIGURE 11, on the other hand, illustrates the

conversion of AIS signal 130 to "signal fail,, signals

132 and 134. SF message 132 goes to origin node 42, at

which point it is reconverted to AIS message 132.

Next, signal 134 passes through tandem node 46 en route

20   to destination node 48,-which reconverts SF message 134

to AIS message 130.

FIGUREs 10 and 11, therefore, illustrate how the

DS-3 standard specifies operations within the

restoration subnetwork. For a DS-3 path including

25   origin node 42 and destination node 48, with one or

more tandem nodes 44, 46. Custodial nodes 62 and 64 are

on each side of the link failure 66. AIS signal 130 is

a DS-3 standard signal that indicates that there is an

alarm downstream. Moreover, AIS signal 130 could

actually be several different signals. AIS signal 130 propagates downstream so that every node sees exactly the same signal.

With AIS signal 130, there is no way to determine which is a custodial node 62, 64 and which is the tandem node 44, 46. This is because the incoming signal looks the same to each receiving node. The present embodiment takes this into consideration by converting AIS signal 130 to a signal fail or SF signal 132. When tandem node 46 sees SF signal 134, it propagates it through until it reaches destination node 48 which converts SF signal 134 back to AIS signal 130.

Another signal that may propagate through the restoration subnetwork 40 is the ISF signal. The ISF signal is for a signal that comes into the restoration subnetwork and stands for incoming signal fail. An ISF signal occurs if a bad signal comes into the network. if it comes in as an AIS signal, there is the need to distinguish that, as well. In the SONET standard there is already an ISF signal. The present invention adds the SF signal, as previously mentioned. In the DS-3 standard, the SF signal already exists. The present invention adds the ISF signal to the DS-3 standard. Consequently, for operation of the present invention in the DS-3 standard environment, there is the addition of the ISF signal. For operation in the SONET standard environment, the present invention adds the SF signal. Therefore, for each of the standards, the present invention adds a new signal.

To distinguish whether an incoming non-traffic
signal received by a node has been asserted due to an
alarm within a DRA-controlled network, a modified DS-3
idle signal is propagated downstream in place of the

5   usual Alarm Indication Signal (AIS). This alarm-
produced idle signal differs from a normal idle signal
by an embedded messaging in the C-bit maintenance
channel to convey the presence of a failure within the
realm of a particular network. The replacement of AIS

10 with idle is done to aid fault isolation by squelching
downstream alarms. Upon leaving the network, such
signals may be converted back into AIS signals to
maintain operational compatibility with equipments
outside the network. A comparable technique is

15 performed in a SONET network, where STS-N AIS signals
are replaced with ISF signal and the ZS byte conveys
the alarm information.

Another aspect of the present invention is the
ability to manage unidirectional failures. In a

20 distributed restoration environment, failures that
occur along one direction of a bidirectional link are
handled by first verifying that the alarm signal
persists for a period of time and then propagating an
idle signal back along the remaining working direction.

25 This alarm produced idle signal differs from a normal
idle signal by embedded messaging in the C-bit
maintenance channel to convey the presence of a far end
receive failure. In this manner, custodial nodes are
promptly identified and restorative switching is

simplified by treating unidirectional failures as if
they were bidirectional failures.

FIGURE 12 illustrates the broadcast of failure
notification messages from custodial nodes 62 and 64.

5   As FIGURE 12 depicts, custodial node 62 sends a failure
notification to origin node 42, as well as to tandem
node 136. Tandem node 136 further broadcasts the
failure notification message to tandem nodes 138 and
140. In addition, custodial node 64 transmits a

10   failure notification message to tandem node 46, which
further transmits the failure notification message to
destination node 48. Also, custodial node 64
broadcasts the failure notification message to tandem
node 140.

15   FIGURE 13 illustrates the time diagram for the
first iteration following fault isolation. In
particular, FIGURE 13 shows the time diagram for
explore phase 116 and return phase 124 of iteration 1.
FIGURE 14 further illustrates the time diagram for the

20   completion of iteration 1 and a portion of iteration 2.
As FIGURE 14 indicates, iteration 1 includes explore
phase 116, return phase 124, max flow phase 142 and
connect phase 144. Max flow phase 142 includes a
single step 146. Note that connect phase 144 of

25   iteration 2 shown by region 148 includes six steps, 150
through 160, and occurs simultaneously with explore
phase 162 of iteration 2. Note further that return
phase 164 of iteration 2 also includes six steps 166
through 176.

Each iteration involves explore, return, maxflow, and connect phases. The restored traffic addressed by connect message and the remaining unrestored traffic conveyed by the explore message are disjoint sets.

5 Hence, there is no conflict in concurrently propagating or combining these messaging steps in a synchronous DRA process. In conjunction with failure queuing, this practice leads to a restoration process that is both reliably coordinated and expeditious.

10 The iterations become longer in duration and include more steps in subsequent iterations. This is because with subsequent iterations, alternate paths are sought. A path has a certain length in terms of hops. A path may be three hops or four hops, for example. In

15 the first iteration, for example, a hop count may be set at three. This, means that alternate paths that are less than or equal to three hops are sought. The next iteration may seek alternate paths that are less than or equal to six hops.

20 Setting a hop count limit per iteration increases the efficiency of the process of the present invention. With the system of the present invention, the number of iterations and the number of hop counts for each iteration is configurable. However, these may also be

25 preset, depending on the degree of flexibility that a given implementation requires. Realize, however, that with increased configurability, increased complexity results. This increased complexity may, in some

instances, generate the possibility for inappropriate
or problematic configurations.

FIGURE 15, for promoting the more detailed
discussion of the explore phase, shows explore phase
116, which is the initial part of the first iteration
114. FIGURE 16 shows restoration network portion 170
to express the idea that a single origin node 42 may
have more than one destination node. In particular,
destination node 180 may be a destination node for
origin node 42 through custodial nodes 62 and 66.
Also, as before, destination node 48 is a destination
node for origin node 42. This occurs because two
working paths, 182 and 184, flow through restoration
subnetwork portion 170, both beginning at origin node
42. During the explore phase, messages begin at the
origin nodes and move outward through the restoration
subnetwork. Each explore message is stored and
forwarded in a loosely synchronized manner.
Accordingly, if a node receives the message in step 1,
it forwards it in step 2. The neighboring node that
receives the explore message in step 1 transmits the
explore message to its neighboring node in step 2.
Because the present invention employs loose
synchronization it does not matter how fast the message
is transmitted from one neighbor to another, it will be
sent at the next step irrespectively.

If the explore phase is three steps long, it may
flood out three hops and no more. The following
discussion pertains to a single origin-destination

pair, but there may be other origin/destination pairs performing the similar or identical functions at the same time within restoration subnetwork 40. If two nodes send the explore message to a neighboring node,

5 only the first message received by the neighboring node is transmitted by the neighboring node. The message that is second received by the neighboring node is recognized, but not forwarded. Accordingly, the first node to reach a neighboring node with an explore

10 message is generally the closest node to the neighboring node. When an explore message reaches the destination node, it stops. This step determines the amount of spare capacity existing in the restoration subnetwork between the origin node and the destination

15 node.

Because of loose synchronization, the first message that reaches origin node 42 and destination node 48 will be the shortest path. There are no race conditions within the present invention's operation.

20 In the explore message, the distance between the origin node and destination node is included. This distance, measured in hops, is always equal to or less than the number of steps allowed for the given explore phase. For example, if a destination node is five hops from

25 the origin node by the shortest path, the explore phase with a three hop count limit will never generate a return message. On the other hand, an explore phase with a six hop count limit will return the five hop count information in the return message.

In the explore message there is an identification
of the origin-destination pair to identify which node
sent the explore message and the destination node that
is to receive the explore message. There is also a
request for capacity. The message may say, for
example, that there is the need for thirteen DS-3s,
because thirteen DS-3s failed. In practice, there may
be not just DS-3s, but also STS-1s, STS-12C's, etc.
The point being, however, that a certain amount of
capacity is requested. At each node that the explore
message passes through, the request for capacity is
noted. The explore phase is over once the
predetermined number of steps have been completed.
Thus, for example, if the explore phase is to last
three steps, at step 4, the explore phase is over.
This provides a well-defined end for the explore phase.

FIGURE 17 illustrates restoration subnetwork 40
for a single-origin destination pair, including origin
node 42 and destination node 48. In restoration
subnetwork 40, origin node 42, at the beginning of the
explore phase, takes step 1 to send an explore message
to tandem node 44, tandem node 46 and tandem node 186.
At step 2, tandem node 46 sends an explore message to
tandem node 188 and to destination node 48. At step 2,
tandem node 44 sends an explore message to tandem node
46, tandem node 46 sends an explore message to tandem
node 188, and to destination node 48, and tandem node
186 sends explore messages to tandem node 46 and to
destination node 48. Note that explore messages at

step 2 from tandem node 44 to tandem node 46 and from tandem node 186 to tandem node 46 are not forwarded by tandem node 46.

FIGURE 18 illustrates the time diagram for the next phase in the restoration process of the present invention, the return phase 24, which during the first iteration, includes three steps, 126, 128 and 129.

FIGURE 19 illustrates the return phase of the present invention, during the first iteration. Beginning at destination node 48, at step 4, return message flows on path 192 to tandem node 46, and on path 190 to tandem node 186. At step 5, the return message flows on path 76 to origin node 42. Also, from tandem node 186, a return message flows to origin node 42.

During the return phase, a return message flows over the same path traversed by its corresponding explore phase, but in the opposite direction. Messages come from the destination node and flow to the origin node. In addition, the return phase messages are loosely synchronized as previously described. The return phase messages contain information relating to the number of spare links available for connecting the origin node to the destination node.

In the return phase, information relating to the available capacity goes to the origin node. Beginning at destination node 48, and continuing through each tandem node 44, 46, 186 en route to origin node 42, the return message becomes increasingly longer. The return

message, therefore, contains information on how much

capacity is available on each span en route to the

origin node. The result of the return message received

is the ability to establish at the origin node a map of

5   the restoration network showing where the spare

capacity is that is useable for the restoration.

FIGURE 20 illustrates tandem node 44, that

connects to tandem node 46 through span 38. Note that

span 38 includes six links 32, 34, 36, 196, 198 and

10   200. FIGUREs 21 and 22 illustrate the allocation of

links between the tandem nodes 44, 46 according to the

preferred embodiment of the present invention.

Referring first to FIGURE 21, suppose that in a

previous explore phase, span 38 between tandem nodes 44

15   and 46 carries the first explore message (5,3)

declaring the need for four links for node 46, such as

scenario 202 depicts. Scenario 204 shows further a

message (11,2) requesting eight link flows from tandem

node 44, port 2.

20   FIGURE 22 illustrates how the present embodiment

allocates the six links of span 38. In particular, in

response to the explore messages from scenarios 202 and

204 of FIGURE 21, each of tandem nodes 44 and 46 knows

to allocate three links for each origin destination

25   pair. Thus, between tandem nodes 44 and 46, three

links, for example links 32, 34 and 36 are allocated to

the (5,3) origin destination pair. Links 196, 198 and

200, for example, may be allocated to the

origin/destination pair (11,2).

FIGURE 23 illustrates the results of the return
phase of the present invention. Restoration subnetwork
40 includes origin node 42, tandem nodes 208, 210 and
212, as well as tandem node 44, for example. As FIGURE

5   23 depicts, return messages carry back with them a map
of the route they followed and how much capacity they
were allocated on each span. Origin node 42 collects
all the return messages. Thus, in this example,
between origin node 42 and tandem node 44, four links

10   were allocated between origin node 42 and node 208.
Tandem node 208 was allocated ten links to tandem node
210. Tandem node 210 is allocated three links, with
tandem node 17. And tandem node 17 is allocated seven
links with tandem node 44.

15   The next phase in the first iteration of the
process of the present invention is the maxflow phase.
The maxflow is a one-step phase and, as FIGURE 24
depicts, for example, is the seventh step of the first
iteration. All of the work in the maxflow phase for

20   the present embodiment occurs at origin node 42. At
the start of the maxflow phase, each origin node has a
model of part of the network. This is the part that
has been allocated to the respective origin/destination
pair by the tandem nodes.

25   FIGURE 25 illustrates that within origin node 42
is restoration subnetwork model 214, which shows what
part of restoration subnetwork 40 has been allocated to
the origin node 42-destination node 48 pair. In
particular, model 214 shows that eight links have been

allocated between origin node 42 and tandem node 46, and that eleven links have been allocated between tandem node 46 and destination node 48. Model 214 further shows that a possible three links may be

5 allocated between tandem node 46 and tandem node 186.

As FIGURE 26 depicts, therefore, in the maxflow phase 142 of the present embodiment, origin node 42 calculates alternate paths through restoration subnetwork 40. This is done using a maxflow algorithm.

10 The maxflow output of FIGURE 26, therefore, is a flow matrix indicating the desired flow of traffic between origin node 42 and destination node 48. Note that the maxflow output uses neither tandem node 44 nor tandem node 188.

15 FIGURE 27 illustrates a breadth-first search that maxflow phase 142 uses to find routes through the maxflow phase output. In the example in FIGURE 27, the first route allocates two units, first from origin node 42, then to tandem node 186, then to tandem node 46,

20 and finally to destination node 48. A second route allocates three units, first from origin node 42 to tandem node 186, and finally to destination node 48. A third route allocates eight units, first from origin node 42 to tandem node 46. From tandem node 46, these

25 eight units go to destination node 48.

The last phase in the first iteration in the process of the present embodiment includes connect phase 144. For the example herein described, connect phase includes steps 8 through 13 of the first

iteration, here having reference numerals 150, 152,

154, 156, 220 and 222, respectively.

The connect phase is loosely synchronized, as

previously described, such that each connect message

5   moves one hop in one step. Connect phase 144 overlaps

explore Phase 162 of each subsequent next iteration,

except in the instance of the last iteration. Connect

phase 144 distributes information about what

connections need to be made from, for example, origin

10   node 42 through tandem nodes 46 and 186, to reach

destination node 48.

In connect phase 144, messages flow along the same

routes as identified during maxflow phase 142. Thus,

as FIGURE 29 suggests, a first message, M1, flows from

15   origin node 42 through tandem node 186, through tandem

node 46 and finally to destination node 48, indicating

the connection for two units. Similarly, a second

message, M2, flows from origin node 42 through tandem

node 186 and then directly to destination node 48, for

20   connecting a three-unit flow path. Finally, a third

connect message, M3, emanates from origin node 42

through tandem node 46, and then the destination node

48 for allocating eight units. Connect phase 144 is

synchronized so that each step in a message travels one

25   hop.

For implementing the process of the present

invention in an existing or operational network,

numerous extensions are required. These extensions

take into consideration the existence of hybrid

networks, wherein some nodes have both SONET and DS-3 connections. Moreover, the present invention provides different priorities for working paths and different qualities for spare links. Fault isolation presents a particular challenge in operating or existing environments, that the present invention addresses. Restricted reuse and spare links connected into paths are additional features that the present invention provides. Inhibit functions such as path-inhibit and node-inhibit are additional features to the present invention. The present invention also provides features that interface with existing restoration processes and systems, such as coordination with an existing restoration algorithm and process or similar system. To ensure the proper operation of the present invention, the present embodiment provides an exerciser function for exercising or simulating a restoration process, without making the actual connections for subnetwork restoration. Other features of the present implementation further include a drop-dead timer, and an emergency shutdown feature to control or limit restoration subnetwork malfunctions. Additionally, the present invention handles real life situations such as glass-throughs and staggered cuts that exist in communications networks. Still further features of the present embodiment include a hold-off trigger, as well as mechanisms for hop count and software revision checking, and a step timer to ensure proper operation.

FIGUREs 30 through 33 illustrate how the present
embodiment addresses the hybrid networks. A hybrid
network is a combination of asynchronous and SONET
links. Restrictions in the way that the present

5   invention handles hybrid networks include that all
working paths must either be SONET paths with other
than DS-3 loading, or DS-3 over asynchronous and SONET
working paths with DS-3 access/egress ports.
Otherwise, sending path verification messages within

10   the restoration subnetwork 40, for example, may not be
practical. Referring to FIGUREs 30 and 31, restoration
subnetwork 40 may include SONET origin A/E port 42,
that connects through SONET tandem port 44, through
sonnet tandem port 46 and finally to sonnet destination

15   A/E port 48. In FIGURE 31, origin A/E port 42 is a DS-
3 port, with tandem port 44 being a sonnet node, and
tandem port 46 being a DS-3 port, for example. Port
106 of destination node 48 is a DS-3 port. In a hybrid
network, during the explore phase, origin node 42

20   requests different types of capacity. In the return
phase, tandem nodes 44, 46 allocate different types of
capacity.

An important aspect of connect phase 144 is
properly communicating in the connect message the type

25   of traffic that needs to be connected. This includes,
as mentioned before, routing DS-3s, STS-1s, OC-3s, and
OC-12Cs, for example. There is the need to keep track
of all of the implementation details for the different
types of traffic. For this purpose, the present

invention provides different priorities of working paths and different qualities of spare links. With the present embodiment of the invention, working traffic is prioritized between high priority and low priority

5   working traffic.

SONET traffic includes other rules to address as well. For instance, a SONET path may include an OC-3 port, which is basically three STS-1 ports, with an STS-1 representing the SONET equivalent of a DS-3 port.

10   Thus, an OC-3 node can carry the same traffic as can three STS-1. An OC-3 node can also carry the same traffic as three DS-3s or any combination of three STS-1 and DS-3 nodes. In addition, an OC-3 node may carry the same traffic as an STS-3. So, an OC-3 port can

15   carry the same traffic as three DS-3, three STS-1, or one OC-3. Then, an OC-12 may carry an OC-12C. It may also carry the same traffic as up to four OC-3 ports, up to 12 STS-1 ports, or up to twelve DS-3 ports. With all of the possible combinations, it is important to

20   make sure that the large capacity channels flow through the greatest capacity at first.

An important aspect of the present invention, therefore, is its ability to service hybrid networks. A hybrid network is a network that includes both SONET

25   and asynchronous links, such as DS-3 links. The present invention provides restoration of restoration subnetwork 40 that may include both types of links. The SONET standard provides that SONET traffic is backward compatible to DS-3 traffic. Thus, a SONET

link may include a DS-3 signal inside it. A
restoration subnetwork that includes both SONET and DS-
3 can flow DS-3-signals, provided that both the origin
A/E port 42 and the destination A/E port 48 are DS-3

5   ports. If this were not the case, there would be no
way to send path verification messages 104 within
restoration subnetwork 40.

        As with pure networks, with hybrid networks,
explore messages request capacity for network

10   restoration. These messages specify what kind of
capacity that is necessary. It is important to
determine whether DS-3 capacity or SONET capacity is
needed. Moreover, because there are different types of
SONET links, there is the need to identify the

15   different types of format of SONET that are needed. In
the return phase, tandem nodes allocate capacity to
origin-destination pairs. Accordingly, they must be
aware of the type of spares that are available in the
span. There are DS-3 spares and SONET spares.

20   Capacity may be allocated knowing which type of spares
are available. There is the need, therefore, in
performing the explore and return phases, to add
extensions that allow for different kinds of capacity.
The explore message of the present invention,

25   therefore, contains a request for capacity and decides
how many DS-3s and how many SONET links are necessary.
There could be the need for an STS-1, an STS-3C, or an
STS-12C, for example. Moreover, in the return phase it
is necessary to include in the return message the

information that there is more than one kind of
capacity in the network. When traffic routes through
the network it must be aware of these rules. For
instance, a DS-3 failed working link can be carried by

5   a SONET link, but not vice versa. In other words, a
DS-3 cannot carry a SONET failed working path.

FIGUREs 32 and 33 illustrate this feature. For
example, referring to FIGURE 32, origin node 42 may
generate explore message to tandem node 44 requesting

10   five DS-3s, three STS-1s, two STS-3(c)s, and one STS-
12(c)s. As FIGURE 33 depicts, from the return phase,
origin node 42 receives return message from tandem node
44, informing origin node 42 that it received five DS-
3s, one STS-1, one STS-3(c), and no STS-12s.

15   For a hybrid restoration subnetwork 40, and in the
maxflow phase, the present invention first routes OC-
12C failed working capacity over OC-12 spare links.
Then, the max flow phase routes OC-3C, failed working
capacity, over OC-12 and OC-3 spare links. Next, the

20   present embodiment routes STS-1 failed working links
over OC-12, OC-3 and STS-1 spare links. Finally, the
max flow phase routes DS-3 failed working links over
OC-12, OC-3, STS-1, and DS-3 spare links. In the
connect phase, the restoration subnetwork of the

25   present invention responds to hybrid network in a
manner so that tandem nodes get instructions to cross-
connect more than one kind of traffic.

FIGURE 34 relates to the property of the present
invention of assigning different priorities for working

paths, and different qualities for spare links. The present embodiment of the invention includes 32 levels of priority for working paths; priority configurations occur at origin node 42, for example. Moreover, the preferred embodiment provides four levels of quality for spare links, such as the following. A SONET 1 for N protected spare link on a span that has no failed links has the highest quality. The next highest quality is a SONET 1 for N protect port on a span that has no failed links. The next highest quality is a SONET 1 for N protected port on the span that has a failed link. The lowest quality is a SONET 1-for-N protect port on a span that has a failed link.

With this configuration, different priorities relate to working paths, and different qualities for spare links. At some stages of employing the present process, the feature of priority working paths and different quality spare links for some uses of the present process, it is possible to simplify the different levels of priority and different levels of quality into simply high and low. For example, high priority working links may be those having priorities 1 through 16, while low priority working links are those having priorities 17 through 32. High quality spares may be, for example, quality 1 spares, low quality spares may be those having qualities 2 through 4.

With the varying priority and quality assignments, the present invention may provide a method for restoring traffic through the restoration subnetwork.

For example, the present invention may first try to restore high priority failed working links on high-quality spare links, and do this as fast as possible. Next, restoring high-quality failed working links on low-quality spares may occur. Restoring low-priority failed working paths on low-quality spare links occurs next. Finally, restoring low priority failed working paths on high quality spare links.

To achieve this functionality, the present invention adds an extra iteration at the end of normal iterations. The extra iteration has the same number of steps as the iteration before it. Its function, however, is to address the priorities for working paths and qualities for spare links. Referring to FIGURE 34, during normal iterations, the present invention will restore high priority working paths over high-quality spare links. During the extra iteration, as the invention restores high-priority working paths over low-quality spare links, then low-priority working paths over low-quality spare links, and finally low-priority working paths over high-quality spare links. This involves running the max flow algorithm additional times.

The network restoration process of the present invention, including the explore, return, and connect messaging phases may be repeated more than once in response to a single failure episode with progressively greater hop count limits. The first set of iterations are confined in restoring only high priority traffic.

Subsequent or extra iterations may be used seek to restore whatever remains of lesser priority traffic. This approach give high priority traffic a preference in terms of path length.

5   FIGUREs 35-37 provide illustrations for describing in more detail how the present invention handles fault isolation. Referring to FIGURE 35, between tandem notes 44 and 46 appear spare link 92. Between custodial nodes 62 and 64 are working link 18 having

10   failure 66 and spare link 196. If a spare link, such as spare link 196, is on a span, such as span 38 that has a failed working link, that spare link has a lower quality than does a spare link, such as spare link 92 on a span that has no failed links. In FIGURE 35,

15   spare link 92 between tandem notes 46 and 48 is part of a span that includes no failed link. In this example, therefore, spare link 92 has a higher quality than does spare link 196.

     Within each node, a particular order is prescribed

20   for sorting lists of spare ports and lists of paths to restore. This accomplishes both consistent mapping and preferential assignment of highest priority to highest quality restoration paths. Specifically, spare ports are sorted first by type (i.e., bandwidth for STS-12,

25   STS-3, then by quality and thirdly by port label numbers. Paths to be restored are sorted primarily by type and secondarily by an assigned priority value. This quality of a given restoration path is limited by the lowest quality link along the path.

In addition to these sorting orders, a process is
performed upon these lists in multiple passes to assign
traffic to spare ports while making best use of high
capacity, high-quality resources. This includes, for
5   example, stuffing high priority STS-1's onto any STS-
12's that are left after all other STS-12 and STS-3
traffic has been assigned.

Rules determine the proper way of handling
different priorities of working paths and different
10   qualities of spares in performing the restoration
process. In our embodiment of the invention, there may
be, for example, 32 priority levels. The working
traffic priority may depend on business-related issues,
such as who is the customer, how much money did the
15   customer pay for communications service, what is the
nature of the traffic. Higher priority working
channels are more expensive than are lower priority
channels. For example, working are assigned priorities
according to these types of considerations. Pre-
20   determined configuration information of this type may
be stored in the origin node of the restoration
subnetwork. Thus, for every path in the origin node
priority information is stored. Although functionally
there is no difference between a high priority working
25   path and lower priority working path, though higher
priority working paths will have their traffic restored
first and lower priority working paths will be restored
later.

The present embodiment includes four qualities of spare links. Spare link quality has to do with two factors. A link may either be protected or nonprotected by other protection schemes. In light of the priorities of failed working paths and the quality of spare links, the present invention uses certain rules. The first rule is to attempt to restore the higher priority failed working paths on the highest quality spare links. The next rule is to restore high quality failed working paths on both high quality and low quality spares. The third rule is to restore low priority failed working paths on low quality spares. The last thing to do is to restore low priority working paths over high and low quality spares.

The present invention also it possible for a node to know when it is a custodial node. Because there are no keep-alive messages on working links, however, the custodial node does not know on what span the failed link resides. Thus, referring to FIGURE 36, custodial node 64 knows that custodial node 62 is on the other end of spare link 196. The difficulty arises, however, in the ability for custodial nodes 62 and 64 to know that working link 18 having failure 66 and spare link 196 are on the same span, because neither custodial node 62 nor custodial node 64 knows on what span is working link 18.

FIGURE 37 illustrates how the present embodiment overcomes this limitation. Custodial node 64, for example, sends a "I am custodial node,, flag in the

keep alive messages that it sends on spare links, such
as to non-custodial tandem node 46. Also, custodial
node 64 and custodial node 62 both send "I am custodial
node" flags on spare 196, to each other. In the event

5   that the receiving non-custodial node, such as tandem
node 46, is not itself a custodial node, then it may
ignore the "I am custodial node", flag. Otherwise, the
receiving node determines that the failure is on the
link between itself and the custodial node from which

10   the receiving custodial node receives the "I am
custodial node" flag.

There may be some limitations associated with this
procedure, such as it may be fooled by "glass throughs"
or spans that have no spares. However, the worst thing

15   that could happen is that alternate path traffic may be
placed on a span that has a failed link, i.e., a lower
quality spare.

The present embodiment provides this functionality
by the use of an "I am custodial node" flag that

20   "piggybacks" the keep alive message. Recalling that a
custodial node is a node on either side of a failed
link, when the custodial node is identified, the "I am
custodial node" flag is set. If the flag appears on a
spare link, that means that the neighboring link is the

25   custodial node. This means that the node is adjacent
to a failure. If the node receiving the flag is also a
custodial node, then the spare is on the span that
contains the failed link. So, the custodial node that
is sending the flag to the non-custodial node, but not

getting it back from a non-custodial node a flag, this
means that the spare link is not in a failed span.

FIGUREs 38-42 illustrate the restricted re-use
feature of the present invention. The present

5   invention also includes a restricted re-use function.
A recovered link relates to the feature of restricted
re-use. Given a path with a failure in it, a recovered
link may exist between two nodes. The recovered link
is a good link but is on a path that has failed.

10   FIGURE 38 shows restoration subnetwork 40 that includes
origin node 42 on link 18 and through custodial nodes
62 and 64 connects to destination node 48. Failure 66
exists between custodial nodes 62 and 64. The
restricted re-use feature of the present invention

15   involves what occurs with recovered links, such as
recovered link 224.

With the present invention, there are at least
three possible modes of re-use. One mode of re-use is
simply no re-use. This prevents the use of recovered

20   links to carry alternate path traffic. Another
possible re-use mode is unrestricted re-use, which
permits recovery links to carry alternate path traffic
in any possible way. Still another re-use mode, and
one that the present embodiment provides, is restricted

25   re-use. Restricted re-use permits use of recovered
links to carry alternate path traffic, but only the
traffic they carry before the failure.

FIGURE 39 illustrates the restricted re-use
concept that the present invention employs. Link 18

enters origin node 42 and continues through tandem node
226 on link 228 and 230 through custodial node 64
through recovered link 48.

Restricted re-use includes modifications to the
explore and return phases of the present invention
wherein the process determines where recovered links
are in the network. The process finds the recovered
links and sends this information to the origin node.
The origin node collects information about where the
recovered links are in the network to develop a map of
the recovered links in the restoration subnetwork. The
tandem nodes send information directly to the origin
node via the wide are network about where the re-use
links are.

FIGURE 40 through 42 illustrate how the present
embodiment achieves restricted re-use. Referring to
restoration subnetwork portion 40 in FIGURE 40, origin
node 42 connects through tandem node 44 via link 78, to
tandem node 46 via link 82, to tandem node 186 via link
84, and to destination node 48 via link 190. Note that
between tandem node 46 and tandem node 186 appears
failures 66.

To implement restricted re-use in the present
embodiment, during the explore and return phases the
origin node 42 will acquire a map of recovered links.
Thus, as FIGURE 40 shows within origin node 42,
recovered links 232, 234, and 236 are stored in origin
node 42. This map is created by sending in-band
messages, re-use messages, during the explore phase,

along recovered links from the custodial nodes to the
origin and destination nodes, such as origin node 42
and destination node 48. Thus, as FIGURE 41
illustrates, in the explore phase, reuse messages

5   emanate from tandem node 46 to tandem node 44 and from
there to origin node 42. From tandem node 186, the re-
use message goes to destination node 48.

In the return phase, such as FIGURE 42 depicts,
the destination node sends the information that it has

10   acquired through re-use messages to the origin node by
piggybacking it on return messages. Thus, as shown in
FIGURE 42, designation node 48 sends on link 192 a
return plus re-use message to tandem node 46. In
response, tandem node 46 sends a return plus re-use

15   message on link 76 to origin node 42.

With the restricted re-use feature and in the max
flow phase, origin node 42 knows about recovered links
and "pure" spare links. When the origin node runs the
max flow algorithm, the recovered links are thrown in

20   with the pure spare links. When the breadth-first-
search is performed, the present invention does not mix
recovered links from different failed working paths on
the same alternate path.

Another feature of the present invention relates

25   to spare links connected into paths. In the event of
spare links being connected into paths, often these
paths may have idle signals on them or a test signal.
If a spare link has a test signal on it, it is not
possible to distinguish it from a working path. In

this instance, the present invention avoids using spare

links with "working" signals on them

In the max flow phase, the origin has discovered

what may be thought of as pure spare link. The origin

5   node also receives information about recovered links,

which the present invention limits to restricted re-

use. In running the max flow algorithm during the max

flow phase of the present process, the pure spare and

recovered links and used to generate a restoration map

10   of the restoration subnetwork, first irrespective of

whether the links are pure, spare or recovered.

Another aspect of the present invention is the

path inhibit function. FIGUREs 43 and 44 illustrate

the path inhibit features of the present invention.

15   For a variety of reasons, it may be desirable to

temporarily disable network restoration protection for

a single port on a given node. It may be desirable,

later, to turn restoration protection back on again

without turning off the entire node. All that is

20   desired, is to turn off one port and then be able to

turn it back on again. This may be desirable when

maintenance to a particular port is desired. When such

maintenance occurs, it is desirable not to have the

restoration process of the present invention

25   automatically initiate. The present invention provides

a way to turn off subnetwork restoration on a

particular port. Thus, as FIGURE 43 shows, origin node

42 includes path 2 to tandem node 44. Note that no

link appears between node 42 and 44. This signifies

that the restoration process of the present invention
is inhibited along path 240 along origin node 42 and
tandem node 44. Working path 242, on the other hand,
exist between origin node 42 and tandem node 46. Link

5   76 indicates that the restoration process of the
present invention is noninhibited along this path if it
is subsequently restored.

During the path inhibit function, the process of
the present invention inhibits restoration on a path by

10   blocking the restoration process at the beginning of
the explore phase. The origin node either does not
send out an explore message at all or sends out an
explore message that does not request capacity to
restore the inhibited path. This is an instruction

15   that goes to the origin node. Thus, during path
inhibit, the process of the present invention is to
inform origin node 42, for example, to inhibit
restoration on a path by sending it a message via the
associated wide area network.

20   Referring to FIGURE 44, therefore, tandem node 46
sends a path inhibit message to origin node 42. Tandem
node 46 receives, for example, a TL1 command telling it
to temporarily inhibit the restoration process on a
port. It sends a message to origin node 42 for that

25   path via wide area network as arrow 246 depicts.

Tandem node 46 sends inhibit path message 246 with
knowledge of the Internet protocol address of its
source node because it is part of the path verification
message. There may be some protocol involved in

performing this function. This purpose would be to cover the situation wherein one node fails while the path is inhibited.

Another feature of the present invention is that it permits the inhibiting of a node. With the node inhibit function, it is possible to temporarily inhibit the restoration process of the present invention on a given node. This may be done, for example, by a TL1 command. A node continues to send its step-complete messages in this condition. Moreover, the exerciser function operates with the node in this condition.

To support the traditional field engineering use of node port test access and path loopback capabilities, the restoration process must be locally disabled so that any test signals and alarm conditions may be asserted without triggering restoration processing. According to this technique as applied to a given path, a port that is commanded into a test access, loopback, or DRA-disabled mode shall notify the origin node of the path to suppress DRA protection along the path. Additional provisions include automatic timeout of the disabled mode and automatic loopback detection/restoration algorithm suppression when a port receives an in-band signal bearing its own local node ID.

Direct node-node communications are accomplished through a dedicated Wide Area Network. This approach bypasses the use of existing in-band and out-of-band call processing signaling and network control links for

a significant advantage in speed and simplicity. In
addition, the WAN approach offers robustness by
diversity.

A triggering mechanism for distributed restoration
process applies a validation timer to each of a
collection of alarm inputs, keeps a count of the number
of validated alarms at any point in time, and generates
a trigger output whenever the count exceeds a preset
threshold value. This approach reduces false or
premature DRA triggering and gives automatic protect
switching a chance to restore individual link failures.
It also allows for localizing tuning of trigger
sensitivity based on quantity and coincidence of
multiple alarms.

The preferred embodiment provides a step
Completion Timer in Synchronous DRA. For each DRA
process initiated within a network node, logic is
provided for automatically terminating the local DRA
process whenever step completion messages are not
received within a certain period of time as monitored
by a failsafe timer. Other causes for ending the
process are loss of keep alive signals through an
Inter-node WAN link, normal completion of final DRA
iteration, consumption of all available spare ports, or
an operation support system override command.

Another aspect of the present invention is a
method for Handling Staggered Failure Events in DRA.
In a protected subnetwork, an initial link failure, or
a set of nearly simultaneous failures, trigger a

sequence of DRA processing phases involving message flow through the network. Other cuts that occur during messaging may similarly start restoration processing and create confusion and unmanageable contentions for spare resources. The present technique offers an improvement over known methods. In particular, during explore and return messaging phases, any subsequent cuts that occur are "queued" until the next Explore phase. Furthermore, in a multiple iteration approach, Explore messaging for new cuts is withheld while a final Explore/Return/Connect iteration occurs in response to a previous cut. These late-breaking held over cuts effectively result in a new, separate invocation of the DPA process.

The present invention includes failure notification messages that include information about the software revision and hop count table contents that are presumed to be equivalent among all nodes. Any nodes that receive such messages and find that the local software revision or hop count table contents disagree with those of the incoming failure notification message shall render themselves ineligible to perform further DRA processing. However, a node that notices a mismatch and disable DPA locally will still continue to propagate subsequent failure notification messages.

The present invention provides a way to Audit restoration process data within nodes that include asserting and verifying the contents of data tables

within all of the nodes in a restoration-protected network. In particular, such data may contain provisioned values such as node id, WAN addresses, hop count sequence table, and defect threshold. The method includes having the operations support system disable the restoration process nodes, write and verify provisionable data contents at each node, then re-enabling the restoration process when all nodes have correct data tables.

In a data transport network that uses a distributed restoration approach, a failure simulation can be executed within the network without disrupting normal traffic. This process includes an initial broadcast of a description of the failure scenario, modified DRA messages that indicate they are "exercise only" messages, and logic within the nodes that allows the exercise to be aborted if a real failure event occurs during the simulation.

Another aspect of the present invention is the ability to coordinate with other restoration processes such as, for example, the RTR restoration system. With the present invention, this becomes a challenge because the port that is protected by the restoration process of the present invention is often also protected by other network restoration algorithms.

Another aspect of the present invention is the exerciser function. The exerciser function for the restoration process of the present invention has two purposes. one is a sanity check to make sure that the

restoration process is operating properly. The other
is an exercise for capacity planning to determine what
the restoration process would do in the event of a link
failure. With the present invention, the exerciser

5   function operates the same software as does the
restoration process during subnetwork restoration, but
with one exception. During the exerciser function,
connections are not made. Thus, when it comes time to
make a connection, the connection is just not made.

10   With the exerciser function, essentially the same
reports occur as would occur in the event of a link
failure. Unfortunately, because of restrictions to
inband signaling, there are some messages that may not
be exchanged during exercise that would be exchanged

15   during a real event. For that reason, during the
exercise function it is necessary to provide the
information that is in these untransmittable messages.
However, this permits the desired exerciser function.

Another aspect of the present invention is a

20   dropdead timer and emergency shut down. The drop-dead
timer and emergency shut down protect against bugs or
defects in the software. If the restoration process of
the present invention malfunctions due to a software
problem, and the instructions become bound and aloof,

25   it is necessary to free the restoration subnetwork.
The dropdead timer and emergency shut down provide
these features. The drop-dead timer is actuated in the
event that a certain maximum allowed amount of time in
the restoration process occurs. By establishing a

maximum operational time the restoration network can
operate for 30 seconds, for example, but no more. If
the 30 second point occurs, the restoration process
turns off.

5   An emergency shut down is similar to a drop-dead
timer, but is manually initiated. For example, with
the present invention, it is possible to enter a TL1
command to shut down the restoration process. The
emergency shut down feature, therefore, provides

10   another degree of protection to compliment the drop
dead timer.

       Out-of-band signaling permits messages to be
delivered over any communication channel that is
available. For this purpose, the present invention

15   uses a restoration process wide area network. For
purposes of the present invention, several messages get
sent out of band. These include the explore message,
the return message, the connect message, the step
complete message, as well as a message known as the

20   exercise message which has to do with an exerciser
feature of the present invention. The wide area
network of the present invention operates under the
TCP/IP protocol, but other protocols and other wide
area networks may be employed. In order to use the

25   wide area network in practicing the present invention,
there is the need for us to obtain access to the
network. For the present invention, access to the wide
area network is through two local area network Ethernet
ports. The two Ethernet ports permit communication

with the wide area network. In the present embodiment
of the invention, the Ethernet is half duplex, in the
sense that the restoration subnetwork sends data in one
direction on one Ethernet while information flows to

5   the restoration subnetwork in the other direction on
the other Ethernet port. The wide area network of the
present invention includes a backbone which provides
the high bandwidth portion of the wide area network.
The backbone includes the same network that the

10   restoration subnetwork protects. Thus, the failure in
the restoration subnetwork could potentially cut the
wide area network. This may make it more fragile.

Accordingly, there may be more attractive wide
area networks to use with the present invention. For

15   example, it may be possible to use spare capacity as
the wide area network. In other words, there may be
spare capacity in the network which could be used to
build the wide area network itself. This may provide
the necessary signal flows to the above-mentioned types

20   of messages. With the present invention, making
connections through the wide area network is done
automatically.

For the cross-connects of the present invention,
there is a control system that includes a number of

25   computers within the cross-connect switch. The
crossconnect may include possibly hundreds of
computers. These computers connect in the hierarchy in
three levels in the present embodiment. The computers
that perform processor-intensive operations appear at

the bottom layer or layer 3. Another layer of computers may control, for example, a shelf of cards. These computers occupy layer 2. The layer 1 computers control the layer 2 computers.

5   The computers at layer 1 perform the instructions of the restoration process of the present invention. This computer may be centralized in the specific shelf where all layer 1 computers are in one place together with the computer executing the restoration process

10   instructions. Because the computer performing the restoration process of the present invention is a layer 1 computer, it is not possible for the computer itself to send in-band messages. If there is the desire to send an in-band message, that message is sent via a

15   layer 3 computer. This is because the layer 3 computer controls the local card that includes the cable to which it connects. Accordingly, in-band messages are generally sent and received by layer 2 and/or layer 3 computers, and are not sent by layer 1 computers, such

20   as the one operating the restoration instructions for the process of the present invention.

Fault isolation also occurs at layer 2 and layer 3 computers within the cross-connects. This is because fault isolation involves changing the signals in the

25   optical fibers. This must be done by machines at lower layers. Moreover, a port, which could be a DS-3 port or a SONET port, has a state in the lower layer processors keep track of the port state. In essence, therefore, there is a division of labor between layer 2

and 3 computers and the layer 1 computer performing the instructions for the restoration process of the present invention.

The exemplar telecommunications network of the instant invention, as shown in FIGURE 45, comprises a number of nodes 302-324 each connected to adjacent nodes by at least one working link and one spare link. For example, node 302 is connected to node 304 by means of a working link 2-4W and a spare link 2-4S. Similarly, node 304 is connect to node 306 by a working link 4-6W and a spare link 4-6S. For the sake of simplicity, only the specific links connecting nodes 302-304, 304-306 and 302-310 are appropriately numbered in FIGURE 45. But it should be noted that the working and spare links connecting adjacent nodes can be similarly designated.

For the telecommunications network of FIGURE 45, it is assumed that all of the nodes of the network are provisioned with a distributed restoration algorithm (DRA), even though in practice oftentimes only one or more portions of the telecommunications network are provisioned for distributed restoration. In those instances, those portions of the network are referenced as dynamic transmission network restoration (DTNR) domains.

Also shown in FIGURE 45 is an operation support system (OSS) 326. OSS 326 is where the network management monitors the overall operation of the network. In other words, it is at OSS 326 that an

overall view, or map, of the layout of each node within

the network is provided. OSS 326 has a central

processor 328 and a memory 330 into which data

retrieved from the various nodes are stored. Memory

5   330 may include both a working memory and a database

store. An interface unit, not shown, is also provided

in OSS 326 for interfacing with the various nodes. As

shown in FIGURE 45, for the sake of simplicity, only

nodes 302, 304, 306, and 308 are shown to be connected

10   to OSS 326. Given the interconnections between OSS 326

and the nodes of the network, the goings on within each

of the nodes of the network is monitored by OSS 326.

Each of nodes 302-324 of the network comprises a

digital cross-connect switch such as the 1633-SX

15   broadband cross-connect switch made by the Alcatel

Network System company. Two of such adjacently

connected switches are shown in FIGURE 46. The FIGURE

46 switches may represent any two adjacent switches

shown in the FIGURE 45 network such as for example

20   nodes 304 and 306 thereof. As shown, each of the

switches has a number of access/egress ports 332, 334

that are shown to be multiplexed to a line terminating

equipment (LTE) 336, 338. LTEs 336 and 338 are SONET

equipment having a detector residing therein for

25   detecting any failure of the links between the various

digital cross-connect switches. Again, for the sake of

simplicity, such LTE is not shown to be sandwiched

between nodes 334 and 336, as detection circuits for

interpreting whether a communication failure has

occurred may also be incorporated within the respective
working cards 340a, 340b of node 304 and 342a and 342b
of node 306.

As shown in FIGURE 46, each of the digital

5   cross-connect switches has two working links 344a and
344b communicatively connecting node 304 and node 306,
by means of the respective working interface cards
340a, 340b and 342a, 342b. Also shown connecting node
304 and node 306 are a pair of spare links 346a and

10   346b, which are connected to the spare link interface
cards 348a, 348b and 350a, 350b of node 304 and node
306, respectively. For the FIGURE 46 embodiment,
assume that each of working links 344a, 344b and spare
links 346a, 346b is a part of a logical span 352.

15   Further note that even though only four links are shown
to connect node 304 to node 306, in actuality, adjacent
nodes may be connected by more or less links.
Likewise, even though only four links are shown to be a
part of span 352, in actuality, a span that connects

20   two adjacent nodes may in fact have a greater number of
links. For the instant discussion, assume that working
links 344a and 344b correspond to the working link 4-6W
of FIGURE 45 while the spare links 346a and 346b of
FIGURE 46 correspond to the spare link 4-6S of FIGURE

25   45. For the purpose of the instant invention, each of
the links shown in FIGURE 46 is presumed to be a
conventional optical carrier OC-12 fiber or is a link
embedded within a higher order (i.e., OC-48 or OC-192)
fiber.

66

Focusing onto node 304 for the time being, note
that each of the interfacing card, or boards, of that
digital cross-connect switch such as 340a, 340b, 348a
and 348b are connected to a number of STS-1 ports 352

5   for transmission to SONET LTE 336. Although not shown,
an intelligence such as a processor residing in each of
the digital cross-connect switches controls the routing
and operation of the various interfacing boards and
ports. Also not shown but present in each of the

10   digital cross-connect switches is a database storage
for storing a map which identifies the various sender
nodes, chooser nodes and addresses, which will be
discussed later. The working boards 342a, 342b and the
spare boards 350a, 350b are likewise connected to the

15   access/egress ports 354 in node 306. Further shown in
FIGURE 46 are non-DRA between adjacent nodes 304 and
306.

For the instant invention, the access/egress ports
such as 332 and 334 send their respective port numbers

20   through the matrix in each of the digital cross-
connects to its adjacent nodes. Thus, for the exemplar
interconnected adjacent nodes 304 and 306, ports 352a
and 352b of node 304 are connected to ports 354a and
354c of node 6 by means of working link 344a.

25   Similarly, ports 352e and 352f are interconnected to
ports 354e and 354f of node 306 by way of spare links
346a and 346b, respectively. Thus, if node 304 were to
transmit a signal using spare link 346a to node 306, it
will be transmitting such a message from its port 352e

to spare card 348a, and then onto spare link 346a, so
that the message is received at spare card 350a of a
conventional optical carrier OC-12 fiber or is a link
embedded within a higher order (i.e., OC-48 or OC-192)

5   fiber.

Focusing onto node 304 for the time being, note
that each of the interfacing card, or boards, of that
digital cross-connect switch such as 340a, 340b, 348a
and 348b are connected to a number of STS-1 ports 352

10   for transmission to SONET LTE 336. Although not shown,
an intelligence such as a processor residing in each of
the digital cross-connect switches controls the routing
and operation of the various interfacing boards and
ports. Also not shown but present in each of the

15   digital cross-connect switches is a database storage
for storing a map which identifies the various sender
nodes, chooser nodes and addresses, which will be
discussed later. The working boards 342a, 342b and the
spare boards 350a, 350b are likewise connected to the

20   access/egress ports 354 in node 306. Further shown in
FIGURE 46 are non-DRA between adjacent nodes 304 and
306.

For the instant invention, the access/egress ports
such as 332 and 334 send their respective port numbers

25   through the matrix in each of the digital cross-
connects to its adjacent nodes. Thus, for the exemplar
interconnected adjacent nodes 304 and 306, ports 352a
and 352b of node 304 are connected to ports 354a and
354c of node 306 by means of working link 344a.

68

Similarly, ports 352e and 352f are interconnected to

ports 354e and 354f of node 306 by way of spare links

346a and 346b, respectively. Thus, if node 304 were to

transmit a signal using spare link 346a to node 306, it

5   will be transmitting such a message from its port 352e

to spare card 348a, and then onto spare link 346a, so

that the message is received at spare card 350a of

node 306 and then routed to the receiving port 354e of

node 306. Thus, as long as each of the working links

10   and spare links interconnecting a pair of adjacent

nodes, such as for example nodes 304 and 306 are

operational, when a message is sent between those

nodes, the information relating to the respective

transmit and receiving ports can be collected by the

15   OSS 326 (FIGURE 45) so that a record can be collected

of the various ports that interconnect any two adjacent

nodes.

For the instant invention, the inventors have

seized upon the idea that a topology, or map, of the

20   available spare capacity of the network, in the form of

the available spare links that interconnect the nodes,

can be generated from stored data that is

representative of the different port numbers of the

various nodes to which spare links are connected. In

25   other words, if a message transmitted by one node to

its adjacent node is able to provide OSS 326 a number

of parameters which include for example the ID of the

transmit node, the respective IP (internal protocol)

addresses of the transmit and receiving ports of the

node and the port number from which the message is
transmitted from the node, the OSS can ascertain, from
similar messages that are being exchanged between
adjacent nodes on spare links connecting those adjacent

5   nodes, an overall picture of the spare capacity of the
network.

Simply put, if each of the digital cross-connect
switches in the DRA provisioned network knows what port
number and the node that it is connected to by its

10   spare link, then that node knows how to reroute traffic
if it detects a failure in one of its working links.
And by collecting the information relating to each of
the nodes of the network, the OSS 326 is able to obtain
an overall view of all of the available spare links

15   that interconnect the various nodes. As a consequence,
when a failure occurs at a given working link, OSS 326
can send to the custodial nodes of the failed link a
map of the spare capacity of the network, so that
whichever custodial node designated as the sender or

20   origin node can then use that map of the spare capacity
of the network to begin the restoration process by
finding an alternate route for rerouting the disrupted
traffic.

The structure of the special message to be used

25   for continuously monitoring the available spare
capacity of the network is shown in FIGURE 47. For the
instant invention, this message is referred to as a
"keep alive" message. As shown, this keep alive
message has a number of fields. Field 356 has an 8 bit

message field. For the FIGURE 47 message, the 8 bits
of data can be configured to represent the keep alive
message so that each node in receipt of the message
will recognize that it is a keep alive message for
updating the availability status of the spare link from
which the message is received. OSS 326, on the other
hand, upon receipt of a keep alive message, would group
it with all the other keep alive messages received from
the different nodes for mapping the spare capacity of
the network.

The next field of the message of FIGURE 47 is
field 358, which is an 8 bit field that contains the
software revision number of the DRA being used in the
network. The next field is 360, which is an 8 bit
field containing the node identifier of the
transmitting node. Field 362 is a 16 bit field that
contains the port number of the transmitting node from
which the keep alive message is sent.

The next field of the message is field 364. This
is a 32 bit field that contains the IP address of the
DS3 port on the node that is used for half duplex
incoming messages. The IP address of the DS3 port of
the node that is used for half-duplex outgoing messages
is contained in the 32 bit field 66.

Field 368 is a 1 bit field that, when set,
indicates to the receiving node that the message is
sent from a custodial node for a failure. In other
words, when there is a failure, the custodial node of
the failed link will send out a keep alive message that

informs nodes downstream thereof that the keep alive

message is being sent from a custodial node since a

failure has occurred, and a restoration process will

proceed.

5   The last field of the keep alive message is field

370.   It has 7 bits and is reserved for future usage.

In operation, before any failure is detected, keep

alive messages such as that shown in FIGURE 47 are.

exchanged on the spare links between adjacent nodes

10   continuously. By the exchange of these keep alive

messages, the network is able to keep a tab of the

various available and functional spare links and also

identify the port number of each node from where each

spare link outputs a keep alive message, as well as the

15   port number of the adjacent node to which the spare

link is connected and to which the keep alive message

is received. By collecting the data that is contained

in each of the keep alive messages, a record is kept of

the various nodes, the port numbers, the incoming and

20   outgoing IP addresses of the various spare links that

are available in the network. And from these collected

data, a topology of the available spare capacity of the

network can be generated, by either the OSS 326, or by

each of the nodes, which can have the collected

25   information downloaded thereto for storage. In any

event, a map of the available spare links of the

network is available, so that when a failure does

occur, the custodial nodes of the failure could

retrieve the up-to-date map of the spare capacity of

the network, and based on that, be able to find the

most efficient alternate route for rerouting the

disrupted traffic.

    Given that the instant invention relates to a

5   distributed restoration process, it should be noted

that an OSS is not necessary for storing the topology

of the spare capacity of the network, as each of the

digital cross-connect switches of the network knows

what port number and the nodes that it is connected to

10   by its spare links. Thus, when a failure occurs, each

of the nodes will continue to send the keep alive

message, as the origin node that is responsible for

restoration can built the entire topology of the

available spare links by retrieving the different keep

15   alive messages from the various nodes. Putting it

differently, an origin node, in attempting to determine

the available spare links, only needs to take the sum

of all of the keep alive messages since each node that

has at least one spare link will send a keep alive

20   message to the origin node. And, by retrieving the ID

of the node and the port numbers of the node to which

spare links are connected, the spare capacity of the

network can be ascertained. As a consequence, the map

of the spare link topology becomes available in a

25   distributed matter to the origin node in the instant

invention DRA provisioned network.

    Inasmuch as the present invention is subject to

many variations, modifications and changes in detail,

it is intended that all matter described throughout

this specification and shown in the accompanying

drawings be interpreted as illustrative only and not in

a limiting sense. Accordingly, it is intended that the

present invention be limited only by the spirit and

5   .scope of the hereto appended claims.